

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A method for optimizing a database query plan, comprising the steps of:

identifying a parent node that is a Join node, wherein said parent node has an original first child node and an original second child node, and wherein the parent node comprises a Join node that is selected from the group consisting of a Left Join node, a Right Join node, a Left Exception Join node and a Right Exception Join node;

changing said parent node to an Inner Join node;

substituting a new node as a first child node of the Inner Join node; and

moving the original first child node to be a child node of the new node.

2. (Original) The method according to claim 1, wherein the parent node comprises a Left Join node.

3. (Original) The method according to claim 2, wherein the new node comprises an Outer Join Fan-out node, the original first child node represents a first table having a first plurality of records, and the original second child node represents a second table having a second plurality of records.

4. (Original) The method according to claim 3, wherein when the query plan is executed, the new node is configured to return, for each record of the second plurality of records, one or more matching records from the first plurality of records according to a set of criteria.

5. (Original) The method according to claim 4, wherein the new node is further configured to return, for any record of the second plurality of records for which no matching records exist, a NULL record.

6. (Original) The method according to claim 1, wherein the parent node comprises a Left Exception Join.

7. (Original) The method according to claim 6, wherein the new node comprises an Exception Fan-out node, the original first child node represents a first table having a first plurality of records, and the original second child node represents a second table having a second plurality of records.

8. (Original) The method according to claim 7, wherein when the query plan is executed, the new node is configured to not return any records, for each record of the second plurality of records having one or more matching records from the first plurality of records according to a set of criteria.

9. (Original) The method according to claim 8, wherein the new node is further configured to return, for any record of the second plurality of records for which no matching records exist, a NULL record.

10. (Original) The method according to claim 1, wherein the parent node comprises a Right Join.

11. (Original) The method according to claim 10, wherein the new node comprises an Outer Join Fan-out node, the original first child node represents a first table having a first plurality of records, and the original second child node represents a second table having a second plurality of records.

12. (Original) The method according to claim 11, wherein when the query plan is executed, the new node is configured to return, for each record of the second plurality of records, one or more matching records from the first plurality of records according to a set of criteria.

13. (Original) The method according to claim 12, wherein the new node is further configured to return, for any record of the second plurality of records for which no matching records exist, a NULL record.

14. (Original) The method according to claim 1, wherein the parent node is a Right Exception Join.

15. (Original) The method according to claim 14, wherein the new node comprises an Exception Fan-out node, the original first child node represents a first table having a first plurality of records, and the original second child node represents a second table having a second plurality of records.

16. (Original) The method according to claim 15, wherein when the query plan is executed, the new node is configured to not return any records, for each record of the second plurality of records having one or more matching records from the first plurality of records according to a set of criteria.

17. (Original) The method according to claim 16, wherein the new node is further configured to return, for any record of the second plurality of records for which no matching records exist, a NULL record.

18. (Currently Amended) An apparatus comprising:

at least one processor;

a memory coupled with the at least one processor; and

an optimizer residing in the memory and executed by the at least one processor, the optimizer configured to identify a Join node in a query plan that interferes with optimization and replace the Join node with an Inner Join node and a new node, wherein the Join node is selected from the group consisting of a Left Join node, a Right Join node, a Left Exception Join node and a Right Exception Join node.

19. (Original) The apparatus according to claim 18, wherein the Join node is one of a Left Join and a Left Exception Join.

20. (Original) The apparatus according to claim 18, wherein the Join node is one of a Right Join and a Right Exception Join.

21. (Original) A program product, comprising:

program code configured upon execution to perform the steps of:

identifying a parent node that is a Join node, wherein said parent

node has an original first child node and an original second child node;

changing said parent node to an Inner Join node;

substituting a new node as a first child node of the Inner Join node;

and

moving the original first child node to be a child node of the new

node.

22. (Original) The program product according to claim 21 wherein the parent node is one of a Left Join node and a Left Exception Join node.

23. (Original) The program product according to claim 21, wherein the parent node is one of a Right Join node and a Right Exception Join node.

24. (Currently Amended) A method performed by a computer executing program code resident on a computer-readable medium for creating at least a portion of a query plan for a Join of a first table and a second table, the method comprising the steps of:

generating Inner Join logic, wherein the Inner Join logic joins the first table and a Fan-out node; and

generating Fan-out node logic, wherein the Fan-out node logic accesses the second table according to a set of selection criteria.

25. (Original) The method of claim 24, wherein the step of generating Inner Join logic includes the steps of:

generating an Inner Join parent node;

generating a first child node of the Inner Join parent node, said first child node comprising the first table; and

generating a second child node of the Inner Join parent node, said second child node comprising the Fan-out node.

26. (Original) The method according to claim 25, wherein the Join is one of a Left Join, Right Join, Left Exception Join, and Right Exception Join.

27. (Original) The method according to claim 25, wherein the step of generating Inner Join logic includes the steps of:

identifying a parent node that is the Join, wherein said parent node has an original first child node representing the first table and an original second child node representing the second table;

changing said parent node to an Inner Join node; and

substituting the Fan-out node as a first child node of the Inner Join node.

28. (Original) The method according to claim 27, wherein when the query plan is executed, the Fan-out node is configured to return:

a) for each record in the first table, one or more matching records from the second table according to the set of selection criteria, and

b) for any record of the first table for which no matching records exist, a NULL record.

29. (Original) The method according to claim 27, wherein when the query plan is executed, the Fan-out node is configured to:

a) not return any records, for each record in the first table, having one or more matching records from the second table according to the set of selection criteria, and

b) for any record of the first table for which no matching records exist, a NULL record.